Our last meeting was held on February 10th. Attending were Lee Gayman, Ed Kapp, Doug Harter and myself. We used various software on two QLs, a 2068 and a Z88. Some QL programs were copied. The 2068 was opened up and some pots. were adjusted to optimize the picture for a color TV. Also some upcoming computer events were discussed such as the Capitolfest and two local Computer/Hamfests in York and Lebanon counties.

The next meeting will be held on March 3rd. The April meeting will be back on the second Friday. It will be on April 14th.

We are planning on having a table at the York Winterfest on March 5th. We are also planning on having a table at the Lebanon Hamfest/Computerfest on April 16th.

I visited the Pittsburgh Area Computer Club meeting on the 19th. I demonstrated transfering files from the Z88 to the PC to Tom Heatherington. Tom is thinking of buying a Z88. Several of the PC users were very impressed with the Z88! I hope that they were impressed enough to buy one. Judy Muir got a letter from Ken Crack who is in the U.K. Ken mentioned that Sir Clive is getting married. We may have a write up on this in a future issue.

I have decided to make this newsletter bimonthly. I see that many of the other TS newsletters are doing so. It has become very difficult getting enough material to put a newsletter together every month.

N E X T   M E E T I N G

FRIDAY MARCH 3rd

CAMP HILL MALL

COMMUNITY ROOM

6:30 to 9:30 PM

David Bennett 329 Rear Walton St.
Lemoyne, PA 17043   717-774-7531

Here are some interesting comments on the Sinclair MSDOS and other machines that I found in the BoSTUG newsletter.

SINCLAIR MS-DOS?

There has been a lot of loose talk floating around the Sinclair Community in North America about an MS-DOS machine called a Sinclair.

Indeed these machines are sold by Sharp's, the venerable dealer in Sinclair QL's and Cambridge Z-88 computers.

But do you really think that Sir Clive would intentionally have anything to do with MS-DOS? What happened is Amstrad got the logo when it bought Sinclair Research. The thing with a Sinclair logo is only an Amstrad MS-DOS machine.

As to its virtues it appears priced to compete well with other 8080 clones and the word is that it is good value. Generally, it comes bundled with a batch of useful software. If you like the Psion programs for the QL, you can get them with enhancements as PC Exchange for under $100.

Truely, though, there remains the continuing question of after Sinclair where do you go?

If you are in love with the 68000 processor, there are several options at different prices.

The Atari ST is a popular choice for some, and the AMIGA is good. There is a QL emulator board for the Atari, at about the price of the Atari, so you can keep on running QL software.

If graphics is your game and you have the folding stuff, you can stay in 68000-land with a MAC.

These computers have one great drawback. They are not black. The obvious choice then? Steve Job's NeXT. It has the 68030 microprocessor, multi-tasks and is black. By the time your QL wears out it should come down in price to something reasonable. All that's needed is for it to have a QL emulator built in.

ADDRESS - QL Archive Program

I thought I would print the QL Archive program that I use to manage the User Group and Exchange Newsletters.

The original program appeared in the Dec 1985/Jan 1986 issue of ZX COMPUTING and was written by David Nowotnik. I also added some procedures from the QL User Guide. Some of these were modified for my use. I also added a few of my own procedures. I find that this program works extremely well.

You probably do not have a User Group to manage so you can modify the program for you own use. Just put in your own fields in the make_file Procedure and modify or delete some of the other Procedures. I modified the program for a records (musical records that is!) file program.

           - David Bennett

```
proc bye
close
print "bye"
endproc
proc change
seek
print at 10,10;"Delete or modify?
(d/m)"
let t$="a"
while not (t$="d" or t$="m")
let t$=getkey()
endwhile
find a$
while found()
display
sprint
if t$="d":remove: endif
if t$="m": alter : endif
continue
endwhile
endproc
```

```
proc device
cls
print "Screen or Printer? ";
let dev$=lower(getkey())
print dev$
endproc
proc doline;x$
if dev$="s": print x$: endif
if dev$="p": lprint x$: endif
endproc
proc initialise
mode 1,6
open "address"
endproc
proc label
rem ***** print labels *****
doline;" "
doline;firstname$+" "+surname$+"
"+dues_date$
doline;street$
doline;town$+", "+state$+"   "+zip$
doline;" "
doline;" "
endproc
proc make_file
create "address"
surname$
firstname$
street$
town$
state$
zip$
telephone$
computer$
interest$
dues_paid$
dues_date$
endcreate
endproc
```

```
proc menu
let s=0
while s=0
title
print at 4,5;"1. Enter new record"
print at 6,5;"2. Search file"
print at 8,5;"3. Delete/modify an
entry"
print at 10,5;"4. Print Labels"
print at 12,5;"5. Print my label"
print at 14,5;"6. Exit file"
print at 14,24;"SELECT 1,2,3,4,5 OR
6"
let s=val(getkey())
if s=1: insert : endif
if s=2:search_file: endif
if s=3:change: endif
if s=4:device:prlabel: endif
if s=5:device:prdave: endif
if s=6: close : endif
if s<>6: let s=0: endif
endwhile
endproc
proc prdave
rem *** print my address label ***

cls
input "How many copies do you want?
";c
let x=c
while x>0
doline;" "
doline;"HATS c/o David Bennett"
doline;"329 Rear Walton Street"
doline;"Lemoyne, PA   17043"
doline;" "
doline;" "
let x=x-1
endwhile
endproc
proc prlabel
cls
all
label
endall
endproc
proc remove
print at 14,5;"Do you want to
delete this record? (y/n)"
yes_or_no
if p$="y"
print at 14,5;"          Are you
sure? (y/n)          "
yes_or_no
if p$="y": delete : cls : return :
endif
else
print at 14,5;"
"

endif
endproc
proc search_file
seek
let n=0
find a$
while found()
let n=n+1
display
sprint
print at 13,4;n
print at 13,10;"PRESS ANY KEY TO
CONTINUE"
let q$=getkey()
print at 13,10;"
"
continue
endwhile
endproc
proc seek
cls :title
input at 8,10;"Enter a search word
- ";a$
endproc
proc start
initialise
menu
endproc
proc title
cls
print at 2,22;"NAME AND ADDRESS
FILE"
endproc
proc yes_or_no
let p$="a"
while not (p$="y" or p$="n")
let p$=getkey()
endwhile
endproc
```

ALTKEYS TO BOOT -or-
    Giving 'fred' the Boot -or-
       Oh No! Not Another BOOT Program!!
_____

For those of you fortunate enough to have Toolkit II but have yet to take advantage of the ALTKEY commands as described in some of Dave Bennett's articles in previous issues of this newsletter, I would like to further convince you that utilizing this function is, in fact, quite worth the effort! When I first tried Dave's Superboot, I said to myself, 'Gee, isn't that neat!'. The problem, however, was that since I use TASKMASTER almost all the time (and it has its own BOOT), I never seemed to have the ALTKEY functions on board when I really needed them. Since then, I have incorporated Dave's Superboot with my TASKMASTER boot and now wouldn't be without the ALTKEY functions (and other things, including an instantly recalled ALTKEY MENU). These are now loaded up automatically when I boot-up TASKMASTER. (I also have SPELLBOUND, so it is really the SPELLBOUND boot that I have modified.)

Here is a list of the things I have added to my BOOT program:

1. Invoke Toolkit II (i.e. 'TK2_EXT')
2. Activate SPEEDSCREEN
   (i.e. '_SPEED 3')
3. Assign 13 ALTKEY designations.
4. Load a previously saved ALTKEY Menu screen into RAM disc for instant recall.
5. Ask if I really want to wait for SPELLBOUND to load ('Hi Rachel'), or go directly to TASKMASTER (as described by Gil Lamb in the July '88 issue of QUANTA).
6. Proceed with the original BOOT program at this point.

In his original SUPERBOOT, Dave incorporated a clock setting routine which I have not included since TASKMASTER has its own routine for doing this.

One of the best reasons why I use the ALTKEY function is that it almost eliminates the need to type in the designated drive device (FLP1_, MDV2_,etc.).
I only recently learned that once Toolkit II is activated, it is not necessary to type in 'LRUN FLP1_fred' because by default, you get the same results by typing 'LRUN fred'. More recent still, I learned that by using the DATA_USE command in Toolkit II, I can change the default device to be any drive I please. So by setting an ALTKEY for each drive, I can enter'ALTKEY 'l''for my left drive(FLP1_), 'ALTKEY 'r'' for my right

(FLP2_), etc. and virtually eliminate the need to find the underline ('_') key!

Another reason I was slow to adopt Dave's program, was that I couldn't remember which ALTKEYs did what! He and I together came up with the RAM disc menu system listed here, and now, by remembering that 'ALTKEY 'a'' brings an instant listing of all the ALTKEY designations, it is quite easy to refresh one's memory as to what is available.

The RAM disc approach requires about 33K of memory (and RAM disc capability), so if you don't have RAM disc, can't spare the memory or if you don't have TASKMASTER or SPELLBOUND, don't despair! You can still enjoy the beauty of automatic ALTKEYs. Simply eliminate the menu portion of the program!

Here are the steps to use in adding these additional options to your existing boot:

a. Invoke Toolkit II ('TK2_EXT')
b. Type in this program as listed and save it as 'fred' ('SAVE dev$_fred'). (Instead of 'dev$', enter the microdrive or disc you are using.)
c. Load your original BOOT program and renumber it with the command 'RENUM 1000,'. (This makes room to MERGE 'fred'.)
d. Enter 'MERGE dev$_fred'.
e. Enter 'SAVE_O dev$_fred'. This overwrites and saves the combined programs.
Note: If you are NOT using the instant menu screen, go to item 'g'.
f. Enter 'SAVESCREEN'. This routine draws the ALTKEY menu on the screen and then saves it to your disc or microdrive. BE SURE THE DEVICE IN LINE 600 MATCHES YOURS.
g. Enter 'RUN'. If you did everything correctly (and I didn't give you misinformation), you should be in operation.
If you entered the menu, try ALTKEY 'a' and see if the menu appears.
h. Try all the ALTKEY functions to see if they work.
i. If all works well, reload 'fred' and save it as 'BOOT'.

In my opinion, the DIRectory and CLS options alone make this program worthwhile. These plus Mike de Sosa's Widescreen (ALT 'w')(from his JOS_MIKE listing) and the DATA_USE option make for a real sophisticated system. Now, if only these options could be incorporated in the power-up defaults!.... Oh well.
                                    -Lee Gayman

```
                    ALTKEY MENU

This program has set up the following ALTERNATE KEY designations as follows:


                    ALT 'z' = CLS #0
                    ALT 'o' = CLS #1
                    ALT 't' = CLS #2
                    ALT 'n' = NEW
                    ALT 'm' = WMON
                    ALT 'v' = WTV
                    ALT 'j' = JOBS
                    ALT 'd' = DIR FLP1_
                    ALT 'f' = DIR FLP2_
                    ALT 'w' = WideScreen

                    ALT 'a' = ALTKEY Menu

                    ALT 'l' = DATA_USE FLP1_
                    ALT 'r' = DATA_USE FLP2_
```

Fig. 1. Altkey MENU which appears instantly with ALTKEY 'a'.

```
100 REMark ALTKEY BOOT PROGRAM - be sure to
    change drive names to fit your system.
110 BEEP 3736,2733
120 wscr
130 PRINT #0;" Loading TOOLKIT II ...."\"
    and also SPEEDSCREEN ...."\"          an
d also Dave Bennett's SUPERBOOT ALTkey Comm
ands (ALT 'a'=MENU)"
140 TK2_EXT:REMark (Invokes Toolkit II)
150 _SPEED 3:REMark (Invokes SPEEDSCREEN)
160 ALTKEY 'z','cls#0',''
170 ALTKEY 'o','cls#1',''
180 ALTKEY 'f','dir flp2_',''
190 ALTKEY 'n','new',''
200 ALTKEY 't','cls #2',''
210 ALTKEY 'd','dir flp1_',''
220 ALTKEY 'm','wmon',''
230 ALTKEY 'v','wtv',''
240 ALTKEY 'j','jobs',''
250 ALTKEY 'a','lbytes ram1_altscn,131072',
''
260 ALTKEY 'l','data_use flp1_',''
270 ALTKEY 'r','data_use flp2_',''
280 ALTKEY 'w', 'WINDOW #0,500,40,4,216','W
INDOW #1,500,216,4,0','WINDOW #2,500,216,4,
0','paper #1,0','paper #2,0','ink #1,7','in
k #2,4','cls #0','cls #1','cls #2','':    R
EMark A REVISED SCREEN CONFIGURATION MODIFI
ED FROM MIKE de SOSA'S 'JOS_MIKE' PROGRAM.
290 COPY flp1_altscn,ram1_altscn:REMark LOA
DS ALTSCREEN MENU INTO RAMDISC FOR INSTANT
RECALL WHEN 'ALTKEY "a"' IS PRESSED.
300 REMark YOU MAY DELETE THE NEXT 4 LINES
IF YOU DO NOT HAVE SPELLBOUND.
310 PRINT #0,"Do you wish to load SPELLBOUN
D? y/n "
320 ds$=INKEYS(-1):PRINT #0,ds$:PAUSE 50:CL
S #0
```

```
330 IF ds$="Y"OR ds$="y":GO TO 350
340 IF ds$="N" OR ds$="n":LRUN flp1_spellbo
ot2
350 DEFine PROCedure altkmenu
360    BEEP 5322,2444
370    BORDER 1,4,2
380    CLS:CLS #0:r:PRINT\ "
                  ALTKEY MENU": INK 4
390    PRINT \"  This program has set up the
 following";:INK 7:PRINT " ALTERNATE KEY";:
INK 4:PRINT " designations as follows:"
400    b$="                       "
410    PRINT\b$;"   ALT";:w:PRINT" 'z'";:g:PR
INT" = CLS #0";\b$;"    ALT";:w:PRINT" 'o'";
:g:PRINT" = CLS #1"\;b$;"     ALT ";:w:PRINT
"'t'";:g:PRINT" = CLS #2"
420    PRINT;b$;"       ALT";:w:PRINT" 'n'";:g
:PRINT" = NEW"\;b$;"       ALT ";:w:PRINT"'m
'";:g:PRINT" = WMON"\b$;"         ALT";:w:PRI
NT" 'v'";:g:PRINT" = WTV";
430    PRINT \b$;"           ALT";:w:PRINT" 'j'
";:g:PRINT" = JOBS"\b$;"           ALT ";:w:P
RINT"'d'";:g:PRINT" = DIR FLP1_"\b$;"
   ALT";:w:PRINT" 'f'";:g:PRINT" = DIR FLP
2_";
440    PRINT\b$;"            ALT";:w:PRINT" '
w'";:g:PRINT" = WideScreen"
450    PRINT\b$;"             ALT";:w:PRINT"
 'a'";:r:PRINT" = ALTKEY Menu":g
460    PRINT\ b$;"              ALT";:w:PRI
NT" 'l'";:g:PRINT " = DATA_USE FLP1_"
470    PRINT b$;"              ALT";:w:PRI
NT" 'r'";:g:PRINT " = DATA_USE FLP2_"
480    DEFine PROCedure w:INK 7:END DEFine
490    DEFine PROCedure g:INK 4:END DEFine
500    DEFine PROCedure r:INK 2:END DEFine
510    DEFine PROCedure p:PRINT;:END DEFine
520    REMark CLS #0:PAUSE(-1):CLS #1
530 END DEFine altkmenu
540 DEFine PROCedure wscr
550    WINDOW #0,500,40,4,216:WINDOW #1,500,
216,4,0:WINDOW #2,500,216,4,0:PAPER #1,0:PA
PER #2,0:INK #1,7:INK #2,4:CLS #0:CLS #1:CL
S #2:REMark THE REVISED SCREEN CONFIGURATIO
N MODIFIED FROM MIKE de SOSA'S 'JOS_MIKE' P
ROGRAM, HERE AS A DEFINED PROCEDURE.
560 END DEFine wscr
570 DEFine PROCedure savescreen
580    altkmenu
590    CLS #0
600    SBYTES flp2_altscn,131072,32768
610 END DEFine savescreen
620 dev$='Flp1_':dic$='Flp1_':REMark   THIS
LINE IS  THE  BEGINNING OF MY 'SPELLBOUND'
BOOT, AND WOULD BE THE PROPER LOCATION FOR
THE BEGINNING OF ANY OTHER BOOT PROGRAM OF
YOUR CHOICE.
```

Z88 COMPUTING by Ian Sinclair
a review

I usually try to avoid computer books which just repeat information in the computers own manual. When the book is bought by mail order it is hard to preview it.

Fortunately I was able to preview this book at a CATS User Group meeting. I decided to buy the book after looking at it.

The Z88 User Guide is a good one particularly for the built in programs. It is a little skimpy on TERMINAL and IMPORT/EXPORT. It also could provide more hardware information.

Z88 COMPUTING is not an advanced guide but it does a good job of supplementing the User Guide.

It covers many of the same topics as the User Guide from the easiest task to the most difficult. It provides additional information and tips which are most useful. With this book you can get more productive use out of your tasks for the Z88.

It also provides some vital additional information about TERMINAL and IMPORT/EXPORT. There is a diagram on a Z88 to Amstrad MSDOS cable. The book also describes several methods of transfering files with TERMINAL and IMPORT/EXPORT and even directly from Pipedream. This information is useful even if you do not have a DOS machine.

The book does not cover use of CLI files. The preface states that; "many users will find no need for them, and only the experienced and skilled user will be able to use them to advantage." I wish that they had included some more information on CLIs. They considerably extend the computer's power. They are particularly useful for basic programming. The book does not cover BASIC at all.

Other than that just about every subject was covered. This book extends the User Guide nicely and provides a good reference source.

— David Bennett

Z88 to QL Connections

A homemade cable and the Q_Link terminal program on the QL works fine transfering files from the Z88 to the QL. To send a file from the Z88 to the QL set identical baud rates on both machines. It is usually 9600. Hit F5 on the QL to open the capture buffer. Send the file with Import-Export. Once the file has transfered, hit F5 on the QL to close the capture buffer. Go into the Q_Link editor with F4. Edit out the Z88 header information in the first line and the 'E' in the last line. Save the file to disk or microdrive.

To send a file from the QL to the Z88, load the file into Q_Link's editor. Set identical baud rates and parameters on both machines. Set the Z88 to receive. Transmit the file with Transmit in the Q_Link Editor. Type 'Y' to send carriage returns and line feeds. Some files from GEnie already have CRs so you would answer 'N' for No. Once you hit 'Y' or 'N' the transfer begins immediately so start receiving on the Z88 just before hitting ENTER on the QL.

Also Pieter van Dijk of the CATS User Group wrote a couple QL programs to transfer files with the Z88. These also work well and filter out some unwanted characters that you do not want to transfer. One of my files will not transfer with Q_Link but does with Pieter's program.

Pieter also provided these cable connections.

```
Z88 DB9                      QL DB9 SER2

   2 ------------------------------------ 3
   3 ------------------------------------ 2
   4 ------------------------------------ 5
   5 ------------------------------------ 4
   7 ------------------------------------ 7
   8 ------------------------------------ 4
```

— David Bennett